

Amendments to the Claims

This listing of claims will replace all prior versions, and listings of claims in the application.

Listing of Claims

- A4
1. (Original) A method for storing a digital value to memory in a pipelined instruction processor, wherein the digital value is read from memory in response to a conditional jump instruction to determine if the condition of the conditional jump instruction is satisfied, the method comprising:
 - 5 generating at least one status bit based on the digital value to be stored, the at least one status bit indicating if a predetermined condition of a conditional jump instruction is satisfied; and
 - 8 storing the digital value and the at least one status bit to memory.
 2. (Original) The method recited in claim 1, wherein the conditional jump instruction reads the digital value and the at least one status bit from memory to determine if the condition of the conditional jump instruction is satisfied.
 - 3 [[2]]. (Currently Amended) The method recited in claim 1, wherein the at least one status bit is read from memory at the same time as the digital value.
 - 4 [[3]]. (Currently Amended) The method recited in claim 1, wherein the memory has one or more addressable locations, and the at least one status bit is stored at the same addressable location as the corresponding digital value.

Application No. 09/727,744
Amendment dated January 20, 2004
Reply to Office Action of October 22, 2003

A4 1 5 [[4]]. (Currently Amended) The method recited in claim 1, wherein the at least
2 one status bit is set high if the digital value is zero.

1 6 [[5]]. (Currently Amended) The method recited in claim 1, wherein the at least
2 one status bit is set high if the digital value is a positive value.

1 7 [[6]]. (Currently Amended) The method recited in claim 1, wherein the at least
2 one status bit is set high if the digital value is negative.

1
1 8 [[7]]. (Currently Amended) The method recited in claim 1, wherein the at least
2 one status bit is set high if the digital value is a non zero value.

1
1 9 [[8]]. (Currently Amended) The method recited in claim 1, wherein the
2 at least one status bit is set high based on the value of the least significant bit of the
3 digital value.

1 10 [[9]]. (Currently Amended) In a pipelined instruction processor that
2 executes instructions including conditional jump instructions, one or more of the
3 conditional jump instructions reading a digital value from memory to determine if the
4 condition of the conditional jump instruction is satisfied, the improvement comprising:

Application No. 09/727,744
Amendment dated January 20, 2004
Reply to Office Action of October 22, 2003

A4

5 status bit generator for generating at least one status bit based on a digital value,
6 the at least one status bit indicating if a predetermined condition of a conditional jump
7 instruction is satisfied; and
8 storing means for storing the digital value and the at least one status bit to the
9 memory.

1 11 [[10]]. (Currently Amended) The pipelined instruction processor recited
2 in claim 10 [[9]], wherein a selected conditional jump instruction reads the digital value
3 and the at least one status bit from memory to determine if the condition of the
4 conditional jump instruction is satisfied.

1 12 [[11]]. (Currently Amended) The pipelined instruction processor recited
2 in claim 10 [[9]], wherein the at least one status bit is read from the memory at the same
3 time as the digital value is read.

1 13 [[12]]. (Currently Amended) The pipelined instruction processor recited
2 in claim 10 [[9]], wherein the memory has one or more addressable locations, and the at
3 least one status bit is stored at the same addressable location as the corresponding digital
4 value.

Application No. 09/727,744
Amendment dated January 20, 2004
Reply to Office Action of October 22, 2003

1 14 [[13]]. (Currently Amended) The pipelined instruction processor recited
A4 2 in claim 10 [[9]], wherein the at least one status bit is set high if the digital value is zero.

1 15 [[14]]. (Currently Amended) The system recited in claim 10 [[9]],
2 wherein the at least one status bit is set high if the digital value is a positive value.

1 16 [[15]]. (Currently Amended) The system recited in claim 10 [[9]],
2 wherein the at least one status bit is set high if the digital value is negative.

1 17 [[16]]. (Currently Amended) The system recited in claim 10 [[9]],
2 wherein the at least one status bit is set high if the digital value is a non zero value.

1 18 [[17]]. (Currently Amended) The system recited in claim 10 [[9]],
2 wherein the at least one status bit is set high based on the value of the least significant bit
3 of the digital value.

1 19 [[18]]. (Currently Amended) In a pipelined instruction processor that
2 executes instructions including conditional jump instructions, one or more of the
3 conditional jump instructions reading a digital value from memory to determine if the
4 condition of the conditional jump instruction is satisfied, the improvement comprising:

Application No. 09/727,744
Amendment dated January 20, 2004
Reply to Office Action of October 22, 2003

5 a plurality of addressable registers, each of the addressable registers storing a
6 value that includes a digital value and at least one jump status bit;
7 logic to access a current instruction, wherein the current instruction includes an
8 address and a corresponding jump field, the address identifies one of the addressable
9 registers and the corresponding jump field identifies a jump status bit of the at least one
10 jump status bits within the identified addressable register;
11 a jump look-ahead controller for generating a jump look-ahead signal using the
12 address that identifies one of the addressable registers and the jump field that identifies a
13 jump status bit within the identified addressable register, the jump look-ahead signal is a
14 function of the identified jump status bit;
15 tracking logic for tracking the addresses of a predetermined number of previous
16 instructions in the pipelined instruction processor and comparing the addresses of each
17 previous instruction to the address of the current instruction to generate a series of jump
18 disable signals; and
19 conflict detection logic for generating a jump early signal using the jump look-
20 ahead signal and the series of jump disable signals, the jump early signal initiates the
21 conditional jump depending on the values of the jump disable signals.

1 20 [[19]]. (Currently Amended) The pipelined instruction processor as
2 recited in claim 19 [[18]], wherein each jump status bit is dependent on the digital value
3 stored in the corresponding addressable register.

Application No. 09/727,744
Amendment dated January 20, 2004
Reply to Office Action of October 22, 2003

1 21 [[20]]. (Currently Amended) The pipelined instruction processor as
2 recited in claim 19 [[18]], further comprising a bit status generator for generating the
3 corresponding jump status bits.

1 22 [[21]]. (Currently Amended) The pipelined instruction processor as
2 recited in claim 19 [[18]], further comprising a prediction logic block responsive to the
3 jump early signal for implementing a prediction algorithm to predict the conditional jump
4 depending on the values of the jump disable signals.

1 23 [[22]]. (Currently Amended) The pipelined instruction processor as
2 recited in claim 19 [[18]], wherein the tracking logic includes a queue for sequentially
3 storing a pre-determined number of instructions prior to sequentially piping the pre-
4 determined number of instructions through a read stage and decode stage in a pre-fetch
5 pipeline.

1 24 [[23]]. (Currently Amended) The pipelined instruction processor as
2 recited in claim 23 [[22]], wherein the pre-determined number of instructions are
3 sequentially piped through an execution pipeline after being piped through a pre-fetch
4 pipeline, the execution pipeline includes a write-back stage.

1

Application No. 09/727,744
Amendment dated January 20, 2004
Reply to Office Action of October 22, 2003

1 25 [[24]]. (Currently Amended) The pipelined instruction processor as
A4 2 recited in claim 24 [[23]], wherein the addressable register is written during the write-
3 back stage.

1 26 [[25]]. (Currently Amended) The pipelined instruction processor as
2 recited in claim 25 [[24]], wherein the execution pipeline further includes an address
3 generation stage, a present address stage, an output operand stage, a capture data stage,
4 and an arithmetic operation stage, all before the write-back stage.

1 27 [[26]]. (Currently Amended) A method for determine if a condition of
2 a conditional jump instruction is satisfied in a pipelined instruction processor, the
3 method comprising:
4 storing a digital value and one or more jump status bits that are based on the
5 digital value in each of a plurality of address locations in an addressable memory;
6 accessing a current instruction, the current instruction having an address and a
7 jump field, the address identifies a selected address location of the addressable
8 memory, and the jump field identifies a selected jump status bit of the selected
9 address location;
10 generating a jump look-ahead signal that is a function of the selected jump
11 status bit read from the selected address location of the addressable memory, the

12 identified jump status bit is accessed using the address and the jump field of the
13 current instruction;
Ay
14 tracking the addresses of a predetermined number of previous instructions in
15 the pipelined instruction processor and comparing the addresses to the address of the
16 current instruction to generate a series of jump disable signals; and
17 generating a jump early signal using the jump-look ahead signal and the series
18 jump disable signals, the jump early signal initiates a conditional jump depending on
19 the value of the jump disable signals.